# *Lab 2 – User Stories*

Date Assigned: Tuesday, August 27

Date due: **Tuesday, August 27 end of lab**

**Learning Objectives**

Upon successful completion of this lab exercise, the student will be able to:

* Understand how to write a user story;
* Break a user story into smaller user stories applying different strategies;
* Write functional and non-functional acceptance criteria for user stories;
* Break a user story into tasks; and
* Estimate the relative size of tasks.

To Start:

1. Download a copy of the Lab 2 file from the Moodle page.
2. Save this document as a Word document named YourUserName\_K30\_L02\_user\_Stories.docx in your 420-K30 folder on your home drive. The document will hold your answers for your lab.
3. Research from class notes and resources, and online resources to help with the questions.
4. Write your answers to the lab questions in the appropriate locations in this file and be sure to save.
5. When you are ready to have your lab marked, notify the professor.
6. When you are finished, submit to Moodle.

To Do:

**Part A – HVK User Stories (Group)**

1. Capture your user stories here:

|  |  |  |
| --- | --- | --- |
| User Story # | User Story | Acceptance Criteria |
| 1 | As I customer I want to be able to create a reservation so that I don’t have to call | The reservation is confirmed after submission. |
| 2 | As a customer I want to be able to delete a reservation so that I don’t have to pay a late fee | The system updates the availability after a reservation is deleted |
| 3 | As an employee I want to be able to create a reservation so that if a customer calls, I can book it | A confirmation email is sent to the customer after the reservation is made by the employee |
| 4 | As an employee I want to be able to delete a reservation so that I can free up space for another customer | The system sends a cancellation email to the customer |
| 5 | As a customer I want to be able to make an account so that I can make a reservation | The customer can log into the system after successful account creation and verification |
| 6 | As an employee I want to be able to allocate runs so that I can see what runs are available | The system updates run availability in real time |
| 7 | As a customer I want to create a pet so that I can reuse info on separate bookings | The system allows multiple pets to be added under the same account |
| 8 | As a customer I want to update a pet so I can keep their information up to date | The system automatically updates the pet’s information across all future reservations |

1. Do your User stories follow the INVEST guidelines? Explain why, why not.

They do follow INVEST guidelines, they are all independent from one another, they can be negotiated and don’t have to be part of the system. They are in valuable language to the product manager, you can estimate how long each will take, and they can all be tested.

1. Do your Acceptance criteria follow guidelines (see rubric)? Explain why, why not.

Yes, because they can be tested, they are clear and concise, easily understood, and provide perspective

**Part B – HVK User Stories Estimation (Group)**

1. Capture your user stories here:

|  |  |  |  |
| --- | --- | --- | --- |
| User Story # | User Story | Estimation points - Poker | Estimation points - RTT |
| 1 | As I customer I want to be able to create a reservation so that I don’t have to call | 20 | 20 |
| 2 | As a customer I want to be able to delete a reservation so that I don’t have to pay a late fee | 2 | 2 |
| 3 | As an employee I want to be able to create a reservation so that if a customer calls, I can book it | 20 | 20 |
| 4 | As an employee I want to be able to delete a reservation so that I can free up space for another customer | 2 | 2 |
| 5 | As a customer I want to be able to make an account so that I can make a reservation | 2 | 3 |
| 6 | As an employee I want to be able to allocate runs so that I can see what runs are available | 5 | 8 |
| 7 | As a customer I want to create a pet so that I can reuse info on separate bookings | 13 | 1 |
| 8 | As a customer I want to update a pet so I can keep their information up to date | 1 | 20 |

1. Analyze- is there a difference between the results of both estimation techniques?

They seem to line up well consistently except for at the end where there is a swap between two user stories. This might be a mistake, or we discussed something during the RTT that we didn’t consider during Planning Poker

1. Analyze – Which estimation technique would you prefer to generate good estimates and why?

Planning Poker – because it is more unbiased than tabletop relative estimation because votes are anonymous until the time of reveal so people who are quieter can’t hide behind copying other people’s estimates.

1. Analyze – Explain how story points and sprint velocity relate

With story points you can know how to spread your tasks through sprints if you know your sprint velocity rate. For example, if your sprint velocity rate is 20, you can go through 20 estimations points of planning poker with confidence knowing that you’re more than likely to be able to get everything done with not too much time left over.

**Part C - Practice with User Stories (individual)**

Write user stories (using the format taught in class) for the following scenarios.

1. For a website that sells books, write a user story that describes the ability of a shopper to be able to see reviews from other shoppers.
   1. First, write a high-level user story for this scenario.

As a shopper, I want to be able to see reviews from other shoppers so that I can make informed decisions before purchasing a book.

* 1. Now break this user story into 3 smaller stories.
     + As a shopper, I want to be able to see the overall rating for each book so I can quickly see its popularity
     + As a shopper, I want to be able to read individual reviews left by other customers so I can see their experience with the book
     + As a shopper, I want to be able to sort reviews so I can have more variety in the types of reviews I am seeing
  2. Write at least 1 functional and 1 non-functional acceptance criteria for each user story in b. Make sure you use different non-functional examples for each.
     + Functional: The system must display the average rating for each book, calculated from all submitted reviews
     + Nonfunctional: The overall rating must be displayed within 1 second of the product page loading
     + Functional: The system must display the individual customer reviews, including the reviewer’s rating, comments, and review date
     + Nonfunctional: The text of the reviews must be in a minimum font size of 14px to ensure accessibility
     + Functional: The system must allow shoppers to sort reviews by highest to lowest, most recent, or most helpful
     + Nonfunctional: The sorting function must complete and update within 2 seconds ensure responsiveness and efficiency
  3. Explain what approach you took for breaking down the user story and why?

I looked at key functionalities that a review system would need to have and turned those into user stories. This makes the tasks clear and allows you to focus on smaller parts at a time. It creates milestones and deliverables and contribute to enhancing the user experience.

1. For a Course Management System (like Moodle), write a user story that describes the capability of a student to be able to access course material and submit assignments.
   1. First, write a high-level user story for this scenario.

As a student, I want to be able to access course material and submit assignments so that I can complete my work on time

* 1. Now break this user story into 3 smaller stories.
     + As a student, I want to be able to view and download course materials so that I can study and complete assignments
     + As a student, I want to be able to upload and submit my assignments online so I can meet deadlines
     + As a student, I want to receive a confirmation after submitting an assignment so that I can sure it was submitted
  2. Write at least 1 functional and 1 non-functional acceptance criteria for each user story in b. Make sure you use different non-functional examples for each.
     + Functional: The system allows students to download each course material
     + Nonfunctional: The downloaded files must be compatible with common devices and software
     + Functional: The system allows but notes late submissions
     + Nonfunctional: The system supports uploads even during peak usage times
     + Functional: The system must display a confirmation message on the screen immediately after successful assignment submission
     + Nonfunctional: The confirmation message must be clearly visible and be displayed for at least 10 seconds
  3. Explain what approach you took for breaking down the user story and why?

I again identified core functions and though about user concerns when using the system like how the student will know his work was submitted. This approach provides clarity and is user focused.

**Part D – User Stories, Tasks, Estimates (individual):**

1. Now using an example from the familiar HVK world, given the high-level user story: *As a clerk I want to be able to search for a customer so I can find their reservation information:*
2. Break this user story into 3 smaller stories.
   1. As a clerk, I want to search for a customer by their name so that I can quickly locate their reservation information
   2. As a clerk, I want to search for a customer by their reservation ID so that I can directly access their specific reservation
   3. As a clerk, I want to filter search results by reservation date so that I can find the correct reservation when multiple entries exist
3. Write at least 1 functional and 1 non-functional acceptance criteria for each user story. (Make sure you don’t just use the same examples from above and for each. Also consider the differences between Black box and White Box testing (i.e. Black box testing – test the system as a whole without knowledge of the internal structure /implementation, White box testing – testing with knowledge /use of the internal implementation/structure).
   1. Functional: The sytem allows the clerk to enter the customer’s full name or partial name into the search bar
   2. Non functional: The system can handlen up to 10000 customer records without performance drop
   3. Functional: The system must display reservation details upon entering a valid ID
   4. Non functional: The system must accept only numeric inputs and validate the reservation ID format
   5. Functional: The filtered results must show only the reservations within the specified dates
   6. Non functional: The systems UI must clearly indicate the active date filter, allowing the clerk to modify or remove it
4. Explain what approach you took for breaking down the user story and why?

For this one I looked at the search functions and all the things the clerk should be able to do with them. This allows robustness in the search system allows for it to be fleshed out and useful to those who use it.

1. Break one user story from a. into tasks and estimate each task using a t-shirt approach, i.e. either assign it an estimate of XS, S, M, L, or XL.

Design the Search Interface (UI) for Name Input

Estimate: M (Medium)

Implement Search Functionality for Full and Partial Name Matches

Estimate: L

Integrate Search Results with Reservation Data

Estimate: M

Optimize Database Queries for Name Search

Estimate: L

Develop and Display Error Handling for No Matches Found

Estimate: S

Test the Search Feature (Black Box and White Box Testing)

Estimate: M

1. Explain how you came up with these estimates? What did you use?

The size of the task was estmated based on the complexity of amount of code I think it would require based on my previous experiences. Some of the tasks have a few layers of complexity will make it so that they take more time than thought at a glance. I was unable to use planning poker or RTT because I was alone but if I had the opportunity I would get a group to help me estimate how long these would take.

**To submit:**

When you have completed the lab save the file and upload the following to Moodle:

* YourUserName\_K30\_L02\_User\_Stories.docx.

**Marking Scheme**

|  |  |
| --- | --- |
|  | Marks |
| Part A |  |
| User stories follow INVEST | 8 |
| Acceptance criteria follows guidelines |  |
| What is acceptable | 4 |
| What is not acceptable | 4 |
| Simple and unambigouous | 4 |
| Functional and non-functional | 4 |
|  |  |
|  |  |
| Part C |  |
| 1 a high-level user story | 2 |
| 1b 3 smaller user stories | 6 |
| 1c 1 functional and 1-non-functional ac for each | 6 |
| 1d approach explained | 2 |
| 2 a high-level user story | 2 |
| 2b 3 smaller user stories | 6 |
| 2c 1 functional and 1-non-functional ac for each | 6 |
| 2d approach explained | 2 |
| Part D |  |
| 1a 3 smaller user stories | 6 |
| 1b 1 functional and 1-non-functional ac for each | 6 |
| 1c approach explained | 2 |
| 1d tasks and estimates for one user story | 6 |
| 1e how did you estimate | 2 |
| Organization – handed into Moodle correctly, proper use of English | 2 |
| Total | 80 |